UNSW
SYDNEY

CANDIDATE

z5383657

TEST

Practice Final Exam

Subject code	--
Evaluation type	--
Test opening time	05.08.2022 08:00
End time	15.08.2022 13:00
Grade deadline	--
PDF created	15.08.2022 12:29

Question	Status	Marks	Question type
i			Information or resources
i			Information or resources
1	Unanswered	0/5	Multiple Choice
2	Correct	5/5	Multiple Choice
3	Correct	5/5	True / False
4	Correct	5/5	Multiple Choice
5	Unanswered	0/5	Multiple Choice
6	Unanswered	0/5	True / False
7	Unanswered	0/5	True / False
8	Unanswered	0/5	True / False
i			Information or resources
9	Unanswered	Manually marked	Essay
10	Unanswered	Manually marked	Essay
11	Unanswered	Manually marked	Essay
12	Unanswered	Manually marked	Essay

1 A generalisation of the Karatsuba trick allows us to multiply two n-bit integers using how many multiplications of n/6-bit integers?

Select one alternative:

- 5
- 7
- 9
- 11 ✔

Maximum marks: 5

2 Suppose you are given an alphabet consisting of several symbols as well as the frequencies with which the symbols appear. The Huffman code is an encoding of each symbol as a binary string which among all prefix codes minimises the expected length of an encoded text. Your task is to find the lengths of all these binary strings.

Suppose the alphabet consists of the following symbols with given frequencies.

Symbol	Frequency
G	20
H	61
I	70
J	2
K	8
L	40
M	24
N	67

What is the set of codeword lengths in a Huffman code for this alphabet?
Select one alternative:

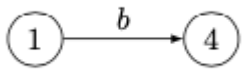
- {2,2,2,3,4,5,6,6}
- {2,2,3,3,3,4,5,5}
- {2,3,3,3,3,3,4,4}
- {2,2,3,3,4,4,4,4}

Maximum marks: 5

3 The Knuth-Morris-Pratt algorithm attempts to find matches for a given pattern by using an automaton. If the pattern has length m, then the automaton will have states numbered 0, ..., m denoting the number of characters matched. From each state, the automaton also records the transition required for each character which could be encountered next.

Consider the finite automaton used by the Knuth-Morris-Pratt algorithm to match the pattern "ababcac".
Five possible transitions are listed below. For each transition, select 'Yes' if it appears in the automaton and 'No' otherwise.

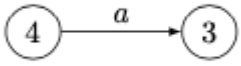
1.



Select one alternative:

- Yes
- No ✓

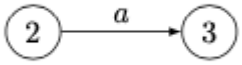
2.



Select an alternative

- Yes ✓
- No

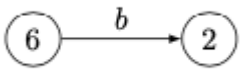
3.



Select an alternative

- Yes ✓
- No

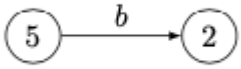
4.



Select an alternative

- Yes ✓
- No

5.

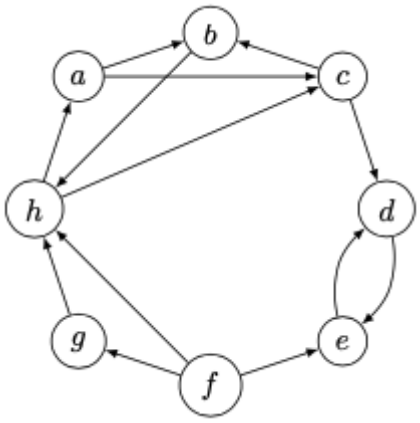


Select an alternative

- Yes
- No ✓

Maximum marks: 5

4 Consider the following graph.



Which of the following sets of vertices is a strongly connected component?
Select one alternative:

- {a,b,h}
- {c,d,e,f,h}
- {a,b,c,h} ✓
- {f,g,h}

Maximum marks: 5

5 Which of the following is the correct asymptotic solution to the recurrence $T(n) = 7T\left(\frac{n}{2}\right) + n^2$?
Select one alternative:

- $T(n) = \Theta\left(n^{\log_2 7}\right)$ ✔
- $T(n) = \Theta\left(n^2 \log n\right)$
- $T(n) = \Theta\left(n^2\right)$
- $T(n) = \Theta\left(n^{\log_2 7} \log n\right)$

Maximum marks: 5

6 Suppose the input to a problem consists of n positive integers, each at most a constant, as well as a positive integer C.

Four runtimes are listed below. For each runtime, select 'Yes' if it belongs to a polynomial time algorithm, or 'No' otherwise.

1. $T(n, C) = n^2 \log_2 C + n$

Select one alternative:

- Yes ✔
- No

2. $T(n, C) = 2^n + \log_2 C$

Select an alternative

- Yes
- No ✔

3. $T(n, C) = nC + 1$

Select an alternative

- Yes
- No ✔

4. $T(n, C) = [\log_2(nC)]^2 + n \log_2 n$

Select an alternative

- Yes ✔
- No

Maximum marks: 5

7 Suppose you are given a polynomial reduction from the Maximum Flow problem to the Travelling Salesman problem, both of which are stated as decision problems and known to be in class NP.

Four statements are listed below. For each statement, determine whether it is true or false.

Throughout this question, you should **only** use the information provided above, and not use any pre-existing knowledge about these problems.

1. If Travelling Salesman is NP-complete, then as a result of this reduction, Maximum Flow is also NP-complete.

Select one alternative:

- True
- False ✔

2. If Maximum Flow is NP-complete, then as a result of this reduction, Travelling Salesman is also NP-complete.

Select an alternative

- True ✔
- False

3. If Maximum Flow is in class P, then as a result of this reduction, Travelling Salesman is also in class P.

Select an alternative

- True
- False ✔

4. If Travelling Salesman is in class P, then as a result of this reduction, Maximum Flow is also in class P.

Select an alternative

- True ✔
- False

Maximum marks: 5

8 You have n rods, each of infinite length. The ith rod costs c_i per metre and has strength s_i per metre, both of which are integers. You can join sections of length x and y (where x and y are positive real numbers) from rods i and j to make a new rod of length $x + y$, cost $c_i x + c_j y$ and strength $s_i x + s_j y$.

Given an integer budget B and an integer length T, your task is to find the maximum strength of a new rod which can be constructed with length exactly T and cost not exceeding B.

In which of the following forms can this problem be written?
Select one alternative:

- Linear Programming ✔
- Integer Linear Programming

Is there a known algorithm to solve Linear Programming in polynomial time?
Select an alternative

- Yes ✔
- No

Is there a known algorithm to solve Integer Linear Programming in polynomial time?
Select an alternative

- Yes
- No ✔

Maximum marks: 5

9 You are given an array A of n positive integers, each at most M. For each pair of distinct indices $1 \leq i < j \leq n$, consider the corresponding sum $A[i] + A[j]$.
Design an algorithm which determines the kth largest of these sums and runs in $O(n \log n \log M)$ time.
You must provide reasoning to justify the correctness and time complexity of your algorithm.

The input consists of the positive integers n, M and k where $k \leq \frac{n(n-1)}{2}$, as well as n positive integers $A[1], \ldots, A[n]$ where each $A[i]$ satisfies $1 \leq A[i] \leq M$.
The output is the kth largest sum $A[i] + A[j]$ where $1 \leq i < j \leq n$.
For example, suppose $n = 4$, $k = 4$ and the array elements are $2, 5, 3, 4$. Going over pairs of distinct indices, we encounter the corresponding sums $5, 6, 7, 7, 8, 9$, so the correct answer is 7. Note that 7 appears twice in the list; it is both the third largest sum and the fourth largest sum.

Hint: for a given positive integer S, can you determine the number of pairs of indices with corresponding sum greater than or equal to S in $O(n \log n)$ time?

Fill in your answer here

Maximum marks: 20

10 There are n monsters planning to take over the city, but only one hero guards the city. The hero has combat effectiveness a_0 and initially has b_0 health points, while the ith monster has combat effectiveness a_i and initially has b_i health points. Both combat effectiveness and health points are positive integers. Both the monsters and the hero die when they reach zero health points or less.

In order to protect the people in the city, the hero will fight monsters until either all monsters are killed or the hero dies. In each fight, the hero can fight any living monster. If the ith monster is selected, then that monster loses a_0 health points and the hero loses a_i health points. After any fight, it is possible that neither the hero nor the monster dies, and it is also possible that both are killed. However, each time the hero kills the selected monster without dying, the hero gains h health points as their success inspires them to keep fighting.

Design an algorithm which determines whether the hero can successfully kill all the monsters (surviving all fights) and runs in $O(n \log n)$ time.

You must provide reasoning to justify the correctness and time complexity of your algorithm.

The input consists of the positive integers n and h, as well as $2n + 2$ positive integers $a_0, b_0, a_1, b_1, \ldots, a_n, b_n$.

The output is either YES or NO.

For example, suppose $n = 2$ and $h = 10$. Suppose the hero has combat effectiveness $a_0 = 5$ and initial health $b_0 = 9$, and that the monsters have:

1. combat effectiveness $a_1 = 3$ and initial health $b_1 = 14$
2. combat effectiveness $a_2 = 4$ and initial health $b_2 = 8$.

The correct answer for this example is YES.

Fill in your answer here

Maximum marks: 20

11 You are given n intervals on an axis. The ith interval $[l_i, r_i)$ has integer endpoints $l_i < r_i$ and has a score of s_i. Your task is to select a set of disjoint intervals with maximum total score. Note that if intervals i and j satisfy $r_i = l_j$ then they are still disjoint.

Design an algorithm which solves this problem and runs in $O(n^2)$ time.

You must provide reasoning to justify the correctness and time complexity of your algorithm.

The input consists of the positive integer n, as well as $2n$ integers $l_1, r_1, \ldots, l_n, r_n$ and n positive real numbers $s_1, \ldots, s_n$.

The output is the set of intervals chosen, organised in any format or data structure.

For example, if $n = 4$ and the intervals are:

1. $l_1 = 0$, $r_1 = 3$, $s_1 = 2$
2. $l_2 = 1, r_2 = 3, s_2 = 1$
3. $l_3 = 2, r_3 = 4, s_3 = 4$
4. $l_4 = 3, r_4 = 5, s_4 = 3$

then you should select only the first and fourth intervals, for a maximum total score of 5. Note that interval 3 is not disjoint with any other interval.

Fill in your answer here

Maximum marks: 20

12 You are given an $n \times n$ grid where some squares have exactly one flower (and the others have none).

Design a polynomial time algorithm which determines whether it is possible to choose exactly n squares so that each chosen square has a flower and no two chosen squares are in the same column or in the same row.

You must provide reasoning to justify that your algorithm is correct and that it runs in polynomial time.

The input consists of the positive integer n, as well as an $n \times n$ array of booleans, where the (i, j) element is TRUE if square (i, j) has a flower and FALSE otherwise.

The output is either YES or NO.

For example, suppose $n = 5$ and the grid is as follows.

T	F	F	T	F
F	T	F	T	F
T	T	F	T	F
F	F	T	F	T
T	F	F	T	F

Then the correct answer is NO.

However, if one extra flower were to be placed at $(3, 3)$ the grid would be as follows.

T	F	F	T	F
F	T	F	T	F
T	T	T	T	F
F	F	T	F	T
T	F	F	T	F

In this case, the correct answer is YES.

Fill in your answer here

Maximum marks: 20